

This volume contains programs which run in the VADCG terminal node controller board. These programs are still in a development stage and may be incomplete or have minor errors. However, most have been used successfully. In any case, they will serve as good examples of how to code the necessary driver software for the board.

PROGRAM SUMMARY

BEACON.ASM - This is the source for a program which will run the board as an HF beacon transmitter. It is set up to run at a speed of 75 Baud. It turns the transmitter on every five minutes and sends a message on the 8273 interface in HDLC format followed by the call sign of the station in CW at 30WPM. One reported bug in this program is that it does not switch to NRZI as stated in the message text. That should be easy to correct.

8255TEST.ASM - Source for a program to exercise the 8255 parallel interface and interrupts. Should be helpful in writing a parallel interface driver program. A change in level of an input line causes the same change in the corresponding output line.

LIP.ASM - This Link Interface Program (LIP) is the protocol driver for the 8273. Basically, it takes the outgoing data blocks from the circular terminal buffer and transmits them. It also receives data on the communications channel and puts it into the circular line buffer and advises the TIP program that they are available. It also provides the protocol for flow control on the link, error recovery, timeouts, retransmissions and handles supervisory frames. It acts in such a way that the TIP program only sees data intended for the terminal node, in proper sequence, free of errors. This particular LIP is set up for a communications channel using the VADCG station node to provide additional function. The initial connection protocol allows the station node to dynamically assign addresses to each terminal node. The protocol allows up to seven frames to be sent from the terminal node in a single transmission before requiring an acknowledgment. The control field format in each frame is very similar to that used in IBM's SDLC line protocol.

TIP.ASM - This is a Terminal Interface Program (TIP). This one is a driver for a 'glass teletype.' Note that besides using the common communications area (CCA), it also uses certain restart (RST) instructions to interface with the LIP. These restarts provide routines to handle the circular buffers and some common routines.

LIPTT.ASM - This is a Terminal to Terminal (TT) LIP designed to provide communications over a channel which does not have a central node set up yet. It was mainly intended to be a stopgap measure but it could be used for off-channel communications after a common channel has been established

in your area. Because a station node is not used the terminal node addresses are hard-assigned. One known problem with this program is that the stack area is too small, so increase it by a few bytes so that you will not get intermittent overlays of data in the CONNECT/DISCONNECT buffer.

TIPTT.ASM - This is a 'glass teletype' driver for a Terminal to Terminal communications channel. It is designed to interface to LIPTT above.

LIB85.LIB - This is the library of new 8085 op codes which is used to assemble some of the above programs. Only the RIM and SIM opcodes are used in these programs. The other codes are not officially supported but I have never found an 8085 that did not have these codes functioning.

STOIC.COM - This is a copy of STOIC containing the words for the VADCG station node controller program. You need an I/O card with an 8273 chip and 8253 interval timers and vectored interrupts to make it run. I am including it for your interest. You can bring it up on your CP/M system and fool around with it. It is very much like FORTH. It will hang up if you try to initialize the station node because the right hardware is not present. If you are interested in setting up the hardware necessary to run this program, I will be glad to give you more information. This version only provides the network service functions of CONNECT, DISCONNECT, SIGNOFF, and USERS. 'Connect <callsign>' establishes a virtual connection between the user and another station on the network. 'Disconnect' breaks the connection and allows reconnection to another station. 'Signoff' removes the user from the network and returns any network resources. 'Users' returns a list of current users and there connection status in table form. In addition, many error messages are returned and logical functions are performed by these network commands. For example, if a connection is established at the time 'signoff' is issued, then network services does an implied disconnect and sends disconnect messages to the corresponding connection. If an attempt is made to connect to a station which is not signed on or to a station which is already connected to someone else then suitable error messages are returned. Other functions are performed but I won't go into further detail at this time.

NOTES ON THE TERMINAL TO TERMINAL PROGRAMS

The following notes were made for the terminal-to-terminal programs but have some application for the regular LIP and TIP programs:

Here are the TIP and LIP programs for providing direct terminal to terminal communication with out the aid of a station node. These programs are working but have not been

given extensive testing so they probably still have a few rough edges. I would appreciate feedback regarding these programs in order to make improvements as I have put several hours of work into them.

GENERAL:

The program LIPTERM.HEX occupies from 0 to 7FFH in memory and fits into the first 2 EPROMS. The TIP program occupies from 800H to BFFH and fits into the third EPROM. Only three EPROMS are required with this software. You will have to burn the LIPTERM program into a couple of EPROMS and your personalized TIP program into another.

PERSONALIZATION:

The TIP program must be set up for your particular situation. The TIP provided is intended as an example but will operate with most ASCII terminals and Teletypes.

Change the following:

1. At label RIMBUF change to your own call sign which must be 6 characters long.
2. At label TERMNO change your terminal's address to any value in the range 1 to 254. It must be unique from any others in your area.
3. Equate the label BAUDRAT to the baud rate constant required by your terminal. Most common speeds are found in the baud rate equates.
4. Equate the label FORMAT to the format constant required for your terminal. Some format constants have been provided but you may have to make up a special one for your terminal. The National data sheet for the INS8250 may help if this is necessary.
5. Make sure LORAM is equated to 1000H and that HIRAM (LIP only) is equated to the highest contiguous ram address installed in the controller. It will be 13FFH if only 2 RAM chips are installed and 1FFFH if all RAM chips are installed.
6. After making these changes, reassemble the TIP and LIP programs. I used MAC to assemble it because there are a couple of simple macros used. You may have to change these macros if another assembler is used.

OPERATION:

When initially turned on, the program will print any correctly received frames. This is called 'MONITOR' mode. No error checking or recovery is done in this mode.

To send a frame of data, type it in and send it off by terminating the line with carriage-return, line feed. Actually only line-feed is required but for proper formatting the carriage return should also be used.

To obtain data integrity in communication one has to establish a connection with a particular terminal. This is done by typing the 6-character call sign of the desired station followed by control-x. The program will then attempt to establish communication. It will try for a

Digitized by the Internet Archive
in 2025 with funding from
Amateur Radio Digital Communications, Grant 151

<https://archive.org/details/vadcg-node-control>

little while and then return to MONITOR mode if it is unsuccessful. If it is successful, it will go into 'CONNECTED' mode and you will see the call sign of the connected station followed by CCCCCC to indicate a successful connection.

Operation in CONNECTED mode is similar to MONITOR mode that the two connected terminals only respond to frames from each other and ignore frames from others on the same channel. In this mode, error recovery is in effect and the two terminals will exchange acknowledgements as required and provide retransmissions of frames if it is necessary.

To disconnect and return to MONITOR mode, press the control-y key. You should see the call sign of the other terminal followed by DDDDDD to indicate you have disconnected and are now back in MONITOR mode.

Sorry for the shortness of this explanation. Please give me a call if you have any questions.

It has been a fair amount of work to develop this software from scratch and to get it working. I would appreciate in return receiving any copies of software based on this work or improvements made in it or bugs found and corrected.

Doug Lockhart, VE7APU

